VeridiumID Administration

Product version 1.8

August 29, 2019

Trademarks

Table of Contents

Administering VeridiumID

You perform the following tasks to manage a VeridiumID deployment.

- Get Started with Administration
- Manage Administrator Accounts
- Manage Users and their Devices
- Monitor User Activity
- Back Up and Restore the VeridiumID database
- View and Manage Configuration Values

Get Started with Administration

Getting started involves the following.

- Accessing the Administration Dashboard
- Best Practices for Administrative Accounts

Accessing the Administration Dashboard

Super-administrators must install the super admin certificate in their browsers to access the administration dashboard with full administrative privileges.

Super administrators have complete access to all operations in the administration dashboard.

Install the Super Admin Certificate

Use this procedure to access the dashboard as a super administrator.

Procedure

1. Provide the admin-cert.p12 certificate and password to one or two administrators who must access the dashboard. The persons who installed the server should have the certificate and password.

   Otherwise, use **scp** (secure copy) or a utility like **WinSCP** to copy the certificate and password to your desktop. These are located here:

   Certificate: **/home/veridiumid/vid_ansible/group_files/dc1/admin-cert.p12**

   Password: **/home/ veridiumid/vid_ansible/group_files/dc1/admin-pass.txt**

2. Import the certificate into your browser.
   - On a Mac, use the keychain app to import the certificate.
   - On other systems with browsers like Chrome, Firefox, Internet Explorer, or Edge, use the browser settings or options to import the certificate.
3. Enter the password when prompted.

Access the Administrator Dashboard

Use this procedure to access the dashboard as a super administrator.

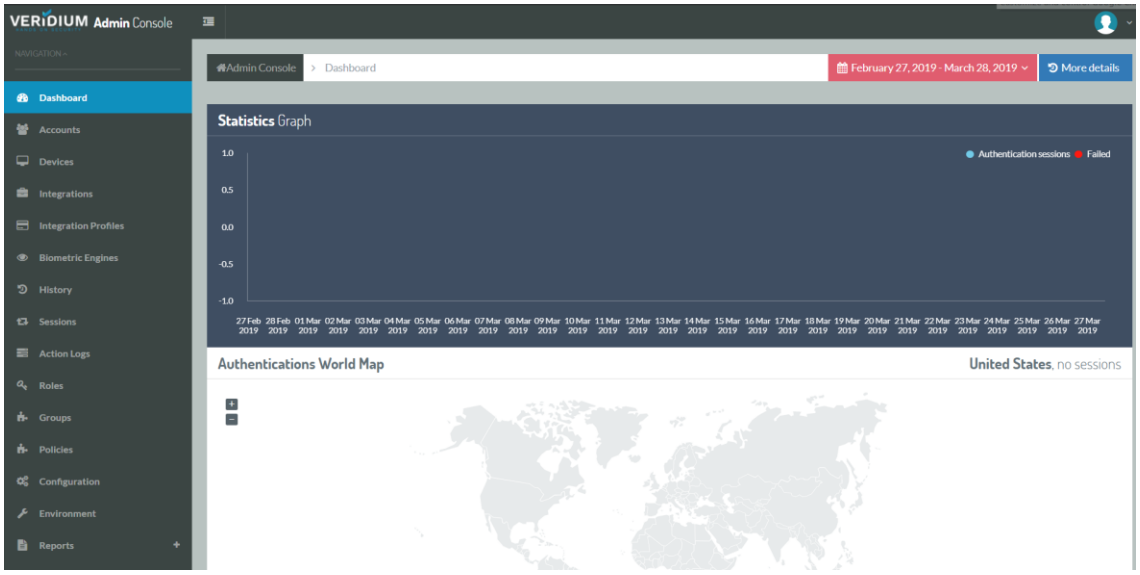
Procedure

- Navigate your browser to the administration UI at a URL like one of these:

  https://vid.example.com/websecadmin/ng/

  https://vid.example.com:9445/websecadmin/ng if using port mapping.

Here is a screen capture of the administration console dashboard.



Best Practices for Administrative Accounts

Super Admin access is convenient for getting started but administrative actions cannot be traced to individual administrators because all Super Admins use the same system account admin@veridiumid.com.

Veridium recommends assigning administrators individual administrative accounts with administrative privileges. You assign administrative privileges such as **App Admins** or **Administrator** to an account to control what administrators can see and do in the console. You use an account's **Privileges** tab to add an administrative group to the account. Logging operations use individual accounts to trace administrative actions to the user that performed the action.

To set up administrative accounts, see Set Up Individual Administrative Accounts.

Manage Administrator Accounts

You add, modify and remove administrator accounts as needed.

VeridiumID Administrative Access Control

VeridiumID supports different administrative groups, roles, and permissions to control access to VeridiumID administrative functions. This table shows the major administrative groups and allowed access. These system groups are fixed and cannot be edited or removed. If you need specialized groups you can add them and assign them to users.

Other groups are available but are reserved for testing and development purposes.

Group	Description
Administrators	This group has extensive rights in the system accessing all accounts and device records for all integrations and all applications. May assign users to administrative groups. Administrators use certificate authentication to access the VeridiumID administration console.
AppAdmins	This group has extensive admin rights limited to the scope of a specific VeridiumID mobile client application. App Admins must enroll using the mobile authentication app for the specific application they manage. AppAdmin users use certificate and biometric authentication to access the VeridiumID console.
TechSupport	This group has rights limited to the scope of a specific VeridiumID mobile client application. Tech Support serves as a help desk group. TechSupport can manually authenticate users, and block, unblock, and remove user accounts. Tech Support users use certificate and biometric authentication to access the VeridiumID console.

Finance, Sales	These groups can run reports limited to the scope of their specific VeridiumID applications. Finance and Sales users use certificate and biometric authentication to access the VeridiumID console.

Scoping Administrative Access

VeridiumID Groups use an administrative user's **Application** to apply administrative scope. An Application consists of these elements:

- Veridium mobile client app
- The integrations the client app is configured to use

Veridium has two kinds of permissions that make up administrative groups.

- The **Cross-Application Administrator** permission is not tied to a specific Application. Users with this permission can see all user accounts in all integrations.
- Application specific permissions (**Application Administrator**, **Technical Support**, and **User can load reports**), limits the administrative scope to user accounts, and data for the integrations that are included in the Application.

Understanding Groups, Roles and Permissions

VeridiumID groups, roles, and permissions control access to individual objects like commands and data objects in the system.

- Permissions control access to individual objects like commands and data objects. All permissions are Application specific except for Cross Application Administrators which is not tied to a specific application.
- Roles are useful collections of permissions.
- Groups contain one or more roles. You add one or more groups to a user account to add needed access permissions to the account owner.

You manually add (or remove) privileges for Administrative users to control their access to operations in the Veridium administration dashboard.

This example describes the relationship of permissions, roles and groups.

Example

You add the **TechSupport** group to a user account for a specific client application scope. This group grants permission to view and manage accounts for users having the same client application scope.

A **TechSupport** user cannot view integrations or other objects outside the scope of his or her application.

- The **TechSupport** group contains the **techsupport** role.
- The **techsupport** role contains only the **Technical Support** permission that limits access to account objects in the scope of the TechSupport user's client application.

For more information about groups, roles and permissions, see Appendix C.

Veridium Group Policies

Use VeridiumID group policies to apply specific behaviors to individual users. You create a policy. Add the policy to a group and apply the group to one or more users. Group policies override global policies and other dynamic rules

For step-by-step procedures, see "Setting Initial Mobile Client Policies."

VeridiumID has one group policy that limits required authentication biometrics to a subset of available types that might differ from Required Biometrics. You give the policy a name like TouchID-Only and apply it to users just like a group.

An example use for this policy could apply to a CEO who spends a lot of time in meetings.

Set Up Individual Administrative Accounts

You must be a **Super Administrator** to initially access the administration dashboard and to begin assigning individual administrative accounts and privileges to other administrative users.

For each administrator, you perform these tasks:

- Create an administrative account
- Assign privileges to the account.
- Create an administration device in the user's account. This generates a certificate that the user imports into their browser.

When you have set several administrators (including yourself) to use individual accounts, you disable Super Administrator access.

Note. If your environment uses a custom authentication and single sign-on solution, you can map permissions stored in LDAP or Active Directory to VeridiumID roles. For details, see Map VeridiumID Groups to Your Active Directory Groups

Create an Administrative User Account

Each VeridiumID administrator must have their own administrative account to access the administration dashboard.

Procedure

1. Access the VeridiumID Administration Interface as a Super Administrator or as Administrator.
2. Click **Accounts**.
3. Click **Add Account.**
4. In the dialog,
    a. select **Account Type** (the relevant Integration),
    b. enter an **External ID** that identifies the account purpose and owner. An example is Admin App_Admin JoseMontoya
    c. enter a unique **Account Name**. This could be the same as the **External ID**.
    d. Click **Create Account**.

    The new account appears in the Account listing.
5. Click 

on the new account.

    If you centrally manage VeridiumID groups using Active Directory, use Active Directory to apply the mapped VeridiumID group to the administrative user. Skip to step 11.

    If you do not centrally manage VeridiumID groups using Active Directory, continue with step 6.
6. Click **Privileges** in the Account Details pane.
7. Scroll down to the Privileges pane.

8. Click in the **Groups list** field.
9. Choose one of these roles from the dropdown list:
    - Administrator
    - AppAdmin
    - Finance (or Sales)
    - TechSupport
10. Click **Save changes**.
11. Click **Devices** in the Account Details pane.
12. Scroll down to view devices listed for the account.
13. Click **+ Create Device** to create an administration device and certificate.
14. Enter the requested information:
    a. **Device External ID**. For instance, **AppAdminDevice**.
    b. **Device Type**. Accept the default **Desktop**.
    c. **Device Name**. For instance, **AppAdminDevice**.
    d. **Device Description**. For instance, **AppAdminDevice.**
    e. If the VeridiumID server is configured to use email, and a valid email address is set for the account, select **Send device certificate by email** and click **Create device**. Otherwise skip to Step f.

       The server sends a certificate and password to the email address set for the Account.

       Instruct the user how to use the certificate and password. See Step 16 for details.
    f. If the server is not set to use email (you did not select **Send device certificate by email**), click **Create device**. The following occurs:
        - A certificate downloads to your browser downloads folder.
        - The certificate password displays.
15. Copy and paste the password into a file for temporary safekeeping.
16. Email the certificate and password to the user with instructions like these:

    You have been added as an Administrator to the VeridiumID server.

    Install the attached certificate in your browser using the password provided.

Access the server administration interface at:

https://veridium-1.example.com/websecadmin/ng/#/app/dashboard

Authenticate using your biometrics on your Veridium Authenticator app when prompted by the VeridiumID server.

If you cannot access the VeridiumID server or if you have any questions, contact VeridiumID server administration.

17. Repeat these steps to add more Administrators as needed.
18. Confirm that added Administrators can access the administration interface.

Modify or Remove an Administrative User

You can add or remove permissions to or from an existing administrator by adding or removing groups in the user account.

Procedure

1. Click **Accounts**.
2. Scroll or search to find the user you want to add as Administrator.

   To search, use the **Accounts Quick Search** function to find a registered user who is to be an Administrator.

   a. Click the **Quick Search** down arrow 

   b. Enter the user **External ID** (or a part of it) and click **Search**.

3. Find the desired External ID and click 

.
4. Click **Privileges** in the Account Details pane.
5. Scroll down to the Privileges pane.
6. To add permissions do the following:
   a. Click in the **Groups list** field.
   b. Click the group you want to add from the dropdown list.
   c. Click **Save changes**.
7. To remove permissions do the following:

a. Click **X** on the groups you want to remove in the **Groups list** field.

b. Click **Save changes**.

8. To completely remove Administrative privileges from a user, do the following:

a. Click **X** on all groups except Guest.

The user becomes a default user.

b. Click **Save changes**.

Disable Super Admin Access

After you have assigned administrative access to enrolled users and confirmed that they can access the VeridiumID administration dashboard, Veridium recommends that you disable Super Admin access.

To prevent lockout, create a new Super Admin certificate but do not distribute it.

Procedure

1. Access the VeridiumID Administration Interface as Administrator.
2. Search for account named **admin@veridiumid.com**.

a. Click **Accounts**.

b. Click the Quick Search down arrow.

c. Scroll down and select the **Is system** checkbox.

d. Click **Search**.

e. Find **admin@veridiumid.com** and click 

.

3. Click **Devices** (tab).
4. Find the Super Admin Device and click 

.
5. Click 

.

A new certificate downloads to your desktop and a password displays. The old certificate is now invalid.

6. Use **scp** (secure copy) or a utility like **WinSCP** to copy the certificate and the password into the **/home/veridiumid/certs** directory on the VeridiumID server. You need the **veridiumid** account password to do this.

You can re-establish certificate access to the VeridiumID console by copying the certificate from **/home/veridiumid/certs** and importing it into your browser, entering the password from the password file when prompted.

Manage Users and their Devices

Administrators perform the following user management tasks.

- Before users enroll, administrators set initial policies to customize enrollment and authentication behaviors.
- Administrators handle user issues like a lost or misplaced phone.

Basic User Account Concepts

The Veridium **Application** is the construct organizing users in the system. An Application contains:

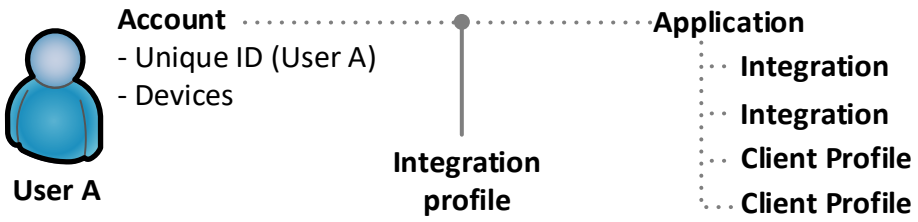
- One or more **integrations**. An integration is a set of rules defining enrollment and authentication policies and workflows. Including multiple integrations under one application allows variable workflows for enrollment and authentication.

  For example, a company might use one integration for most users, another integration with stricter enrollment policies for users who access very sensitive systems, and a third integration for testing purposes.

- One or more client application **profiles**. A profile is the client-side instance of an integration's enrollment and authentication policies and workflows. VeridiumID pushes the user profile to a phone when that user enrolls. VeridiumID pushes other profiles to a phone if, for example, a user is granted certain administrative access to manage VeridiumID users.

When a user enrolls, VeridiumID creates:

- **account** that uniquely identifies the user and user devices (such as mobile phone and desktop) in the VeridiumID system.
- **integration profile** that maps the account to a specific application.



See Appendix B for a more details on integration concepts.

Setting Initial Mobile Client Policies

Before users enroll, you set the following global policy controls on the VeridiumID server that determine how clients enroll and authenticate. Client applications detect these settings at enrollment time and adapt accordingly. These global settings apply to all users in an integration.

- Biometric template storage (client, server, or split) and verification options.
- Which biometrics to enroll and use (4 Fingers TouchlessID, native TouchID).
- Configurable enrollment policies (using LDAP, OTP, Administrator Approval)

Set Biometric Template Storage and Verification Options

Biometric templates enrolled by users can be stored in any of these configurations for mobile and desktop clients.

- Client side verification only (vector is stored on mobile)
- Client side verification with vector split between client and server
- Server side verification with vector split between client and server
- Server side verification only (vector is stored on the server)

Note. Client-side only biometrics FaceID and TouchID are stored and matched only locally on the mobile device as these are controlled by the device hardware.

Use this procedure to set the template verification and storage options for your integration.

Procedure

1. Click **Integrations**.
2. Find the applicable integration and click **Edit**.
3. Scroll to one of the following option lists
   - **Phone Biometry** (for mobile phone Veridium client users)
   - **Desktop Biometry** (for Veridium desktop client users)
4. Choose one of these verification and storage options:
   - **Client side only (vector is stored on mobile)**
   - **Client side verification with vector split**

- **Server side verification with vector split**
- **Server side only (vector is stored on server)**

5. Scroll to the bottom of the dialog and click Save.

Enable Client-Side Only Biometrics

In addition to default biometrics 4 Fingers TouchlessID and Face, VeridiumID supports these client-side only biometrics:

- **FaceID** on iPhoneX
- **TouchID** on iOS
- **Fingerprint** on Android devices.

You can specify these for use on devices that support these biometrics. Users can authenticate using these client-only methods to access Veridium- protected resources.

When users enroll their devices, client-only biometric templates remain on the client and are never stored on the server.

Procedure

1. Click **Biometric Engines**.
2. Click **Add Client Side Biometry**.
3. In the dialog box, enter the following values:
   - Name of the client-side biometric. For example, TouchID.
   - Type of the client-side biometric. For example, TouchID.
   - Version number of the client-side biometric.
   - Number of retries to allow before failing to authenticate.
4. Click **Create Engine**.

The new biometric engine name appears in the list of Biometric Engines.

After You Finish

After you enable a client-only method you must also:

- Set it as an available biometric method.

- Include it in a group policy and apply it to one or more users.

Set Biometrics to Enroll and Use

Use this procedure to set clients to use one or more biometric methods.

Procedure

1. Click **Integrations**.
2. Find the applicable integration and click **Edit**.
3. Scroll to **Available Biometric Methods**. Click in the field and select one or more methods to enable for use.
4. Scroll to **Mandatory Biometric Methods**. Click in the field and select one or more available methods to require for enrollment and use.

   Important. You can add client-side only biometrics such as FaceID or TouchID to **Available Biometric Methods**. You cannot set these to be **Mandatory Biometric Methods** because they might not be provided on all phones in an integration.

5. Scroll to the bottom of the dialog and click 

.

Create a Group Policy for Required Biometric Methods

A group policy can specify a subset of **Available Biometric Methods** and override what is set in **Required Biometric Methods**.

Use a group policy to apply different **Required Biometric Methods** to a subset of users in an integration.

Procedure

1. Click **Policies**.
2. Click **+ Add Policy**.
3. Click in the **Biometrics** field and choose one or more biometric types.
4. Enter a **Policy description** such as **TouchID only**.
5. Enter a unique **Policy name**.
6. Click **Save**.

After You Finish

After you create the group policy you must apply it to one or more users.

Apply a Group Policy to a User

You can apply a group policy to one or more users.

Procedure

1. Access the VeridiumID Administration Console.
2. Click **Accounts**.
3. Scroll or search to find the user you want to add as Administrator. To search:
   a. Click the **Quick Search** down arrow 
   b. Enter the user **External ID** (or a part of it) and click **Search**.
4. Find the desired External ID and click **Details**.
5. Click **Privileges** in the **Account Details** pane.
6. Scroll down to the **Privileges** pane.
7. Click in the **Groups list** field.
8. Choose the Group Policy from the dropdown list.
9. Click **Save changes**.

Set Users' Location Data Parameters

VeridiumID reads end users' geo-location from the phone and uses it to enforce authentication policy. Default, location attributes and values are:

```
"accuracyThreshold": 50,
"countryCodeReplacement": " XB",
"locationAttributeFilter": [
  "accuracy",
  "city",
  "coordinates",
  "countryCode",
  "countryName",
  "district",
  "errorCode",
  "ip",
  "postalCode",
```

```
    "regionCode",
    "regionName",
    "source",
    "street",
    "streetNumber"
]
```

Settable attribute values are

accuracyThreshold: The default is 50 meters. If actual accuracy is worse than the specified value, the phone returns the country code replacement value.

countryCodeReplacement: For locations that do not provide a country code for any reason, the phone returns “XB" indicating cross-border.

You can remove attributes if needed, for instance to satisfy privacy requirements. In this case you must also remove the same attributes from these files:

- **/opt/veridiumid/shibboleth-idp/conf/attribute-resolver.xml**
- **/opt/veridiumid/shibboleth-idp/attributes-filter.xml**

To set location attributes, use the VeridiumID configuration editor to edit **location.json**.

Procedure

1. Access the VeridiumID Administration Console.
2. Click **Configuration** in the Navigation pane.
3. Click **location.json**. The **location** string opens in the editor.
4. Set accuracyThreshold or countryCodeReplacement as needed. The example below uses 100 meters for location accuracy.

```
{
     "accuracyThreshold": 100,
     "countryCodeReplacement": "Remus XB",
     "locationAttributeFilter": [
          "accuracy",
          "countryCode"
     ]
}a
```

5. Scroll down and click **Save**
6. Restart **tomcat** and **websecadmin**.
    a) Sign in to the VeridiumID server console using an ssh client like PuTTY.

b) Enter these commands.

```
# sudo service ver_tomcat restart
# sudo service ver_websecadmin restart
```

Note Omit sudo from the commands in environments that do not use sudo.

Use Friendly Names from Active Directory for Profile Names

Profiles are VeridiumID's way of storing integration policies on the user's phone.

By default in Active Directory environments, VeridiumID tries to name a profile using the Active Directory **displayName** attribute value if that is used.

ADService.json includes the following definition that sets the name:

```
"displayNameAttributes": [
        "displayName"
],
```

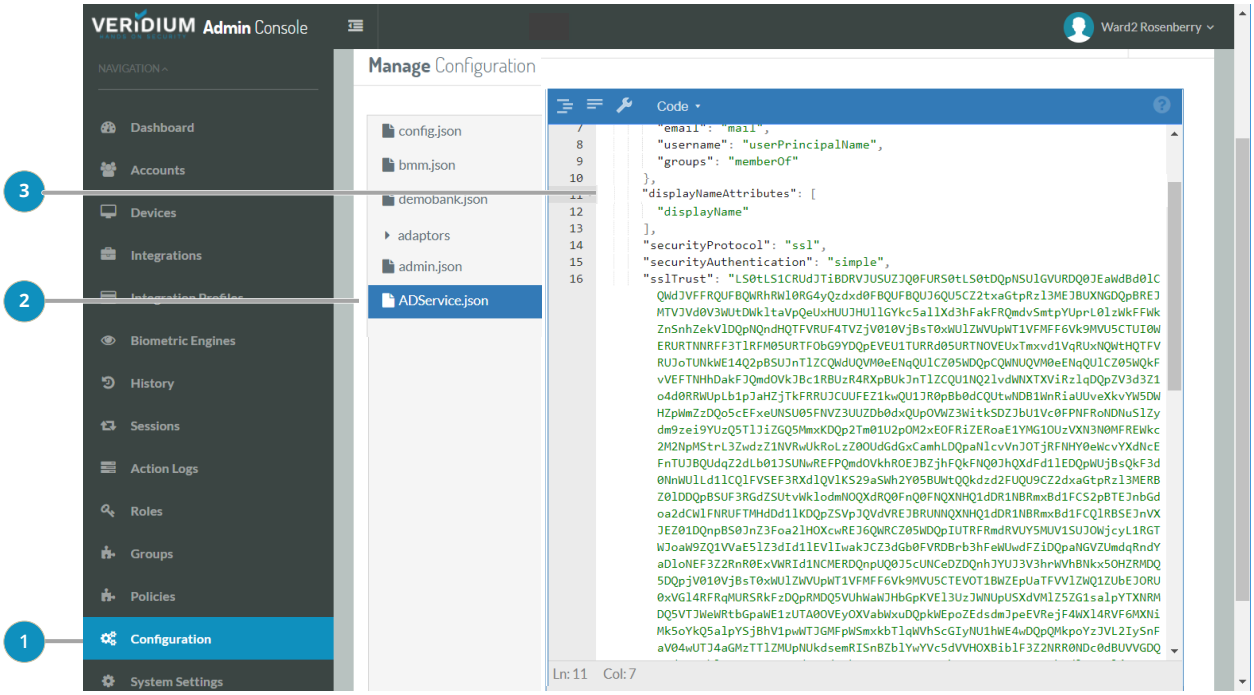
If the **displayName** value is not set in Active Directory VeridiumID uses whatever unique identifier is entered for enrollment such as a login ID or Universal Principal Name, for example.

You can edit **ADService.json** to use other Active Directory attributes like firstName, lastName, department, and so on. Values for these attributes are concatenated, each separated by a space character, and used as the profile name for example, Anna Johnston FINANCE. Here is the coding for the example.

```
"displayNameAttributes": [
        "firstName",
        "lastName",
        "department"
],
```

Procedure

1. Access the VeridiumID Administration Console.
2. Click **Configuration** in the Navigation pane.
3. Click **ADservice.json**.
4. Scroll down or use CTRL-F to find the **displayNameAttributes** section.
5. Edit the lines as needed.
6. Scroll down and click **Save**.



Map VeridiumID Groups to Your Active Directory Groups

Use these procedures map VeridiumID group names you plan to use, to groups contained in Active Directory or your IMS (identity management system). This lets you manage all your groups from a centralized location.

For example, you can map the VeridiumID group **Administrators** to the Active Directory group **Admins**. Alternately you can create a new group in AD like **Veridium-Admins** and map to that name.

Be sure to map any custom groups in VeridiumID, to group names in Active Directory so you can manage them in AD along with the rest of the AD Groups.

You also can set the refresh interval in seconds. This sets how frequently VeridiumID checks Active Directory for changes to group names maintained in Active Directory. The default interval is 300 seconds.

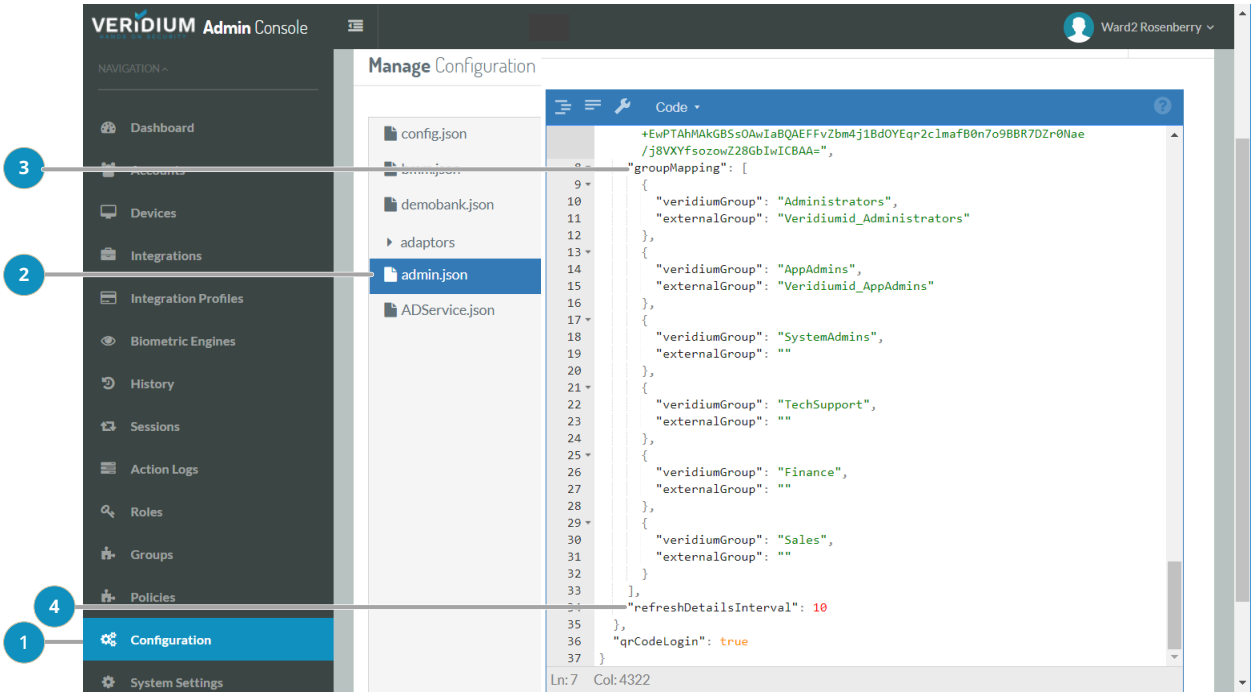
Procedure

1. Access the VeridiumID Administration Console.
2. Click **Configuration** in the Navigation pane.
3. Click **admin.json**.
4. Scroll down to the **groupMapping** section.

5. Enter the group names you want to map to in Active Directory.

   Note: You do not need to map unused VeridiumID groups or VeridiumID group policy groups.

6. If necessary, create these group names in Active Directory.
7. Set **refreshDetailsInterval**. This is the number of seconds VeridiumID waits until it refreshes group names from Active Directory.
8. Scroll down and click **Save.**



Set AD Group Membership to Check before Enrollment

When using the VeridiumID for Active Directory integration, VeridiumID can check that enrolling users are members of specific AD groups before continuing with enrollment. This prevents users who aren't members of the defined groups from enrolling.

One use case for this is limiting enrollments to members of the Finance group.

Procedure

1. Access the VeridiumID Administration Console.

2. Click **Configuration** > **Configuration** in the Navigation pane.
3. Click **Adaptors.**
4. Click **Adv2MultiStepEnrollment.config.json**. The string opens in the editor.
5. Scroll down to the **allowedGroups** descriptors at the root level of the json configuration.

```
"allowedGroups": [
        "admins",
        "helpDesk",
        "FINANCE"
],
```

6. Enter any Active Directory groups you want to check for membership before allowing users to enroll.
7. Scroll down and click **Save**.

Limit how many Phones and PIN types to Allow Per User

By default, VeridiumID does not limit the number of smartphones a user can enroll.

VeridiumID does not limit the PIN types that may be used. For example, VeridiumID can generate a PIN or it can handle a third-party PIN.

Use this procedure to limit the number phones and PIN-types that may be used.

Procedure

1. Access the VeridiumID Administration Console.
2. Click **Configuration >Configuration** in the Navigation pane.
3. Click **config.json**.
4. Scroll down to these parameters to set appropriate values.

Parameter	Description
maxDevicesPerAccount	Sets the number of phones a user can enroll and use to authenticate.
maxPinTokensPerAccount	Sets the number of PIN types to support

Block User after *n* Failed Authentication Attempts

By default, VeridiumID allows up to three consecutive authentication attempts before locking the account to block further attempts to authenticate. A Veridium mobile device locks and displays a message advising blocked users to check with their administrator for more information.

Administrators can unblock a blocked account. See Block, Unblock or Delete an Account.

You can edit **authenticationMaxRetries** in **config.json** to raise or lower the allowed number of consecutive failures.

The range of valid values is 0 (unlimited) to 99.

Procedure

1. Access the VeridiumID Administration Console.
2. Click **Configuration > Configuration** in the Navigation pane.
3. Click **config.json**. The string opens in the editor.
4. Scroll down to **authenticationMaxRetries**.

   ```
   "authenticationMaxRetries": 3,
   ```

5. Set the value as needed.
6. Scroll down and click **Save**.

Set Enrollment Policies

Enrollment policies determine which parameters users must enter when enrolling including user names, domain names, and passwords, and OTP (one-time PIN) codes supplied using SMS. Administrator approvals can also be required.

You set enrollment policies when initially configuring the system.

Enrollment Policy	Enrollment Requirements
OTP only	Users enter their simple user name and domain. Enrollment prompts users to enter an SMS-provided PIN code to complete enrollment. Users do not enter passwords to enroll.
LDAP and OTP	Users enter directory service credentials (user name, password, and domain) for identity validation during enrollment. Users enter an SMS-provided PIN code to complete enrollment.
LDAP only (No OTP)	Users enter directory service credentials (user name, password, and domain) for identity validation to enroll.
LDAP and Administrator Approval	Users enter directory service credentials (user name, and domain) for identity validation during enrollment. Administrators must approve pending enrollments to activate accounts.

Procedure

- Contact your Sales Engineer or Veridium customer support to set or modify an enrollment policy.

Administrator Approval

If your enrollment policy is **LDAP and Administrator Approval**, you must approve each pending user enrollment to activate the account.

Procedure

1. Click **Integration Profiles**.
2. Scroll to find a profile with the status 

.
3. Click **Details**.
4. In the Edit Profile Integration screen, click  (Change status).

5. In the dialog box **Do you want to change the integration profile status?** click the down arrow.
6. Choose **Active** and click **OK**.

   The profile status changes to **Active** and the mobile application activates.

Users Can Enroll and Authenticate on Multiple Devices.

A user with two phones such as a work phone and a personal phone, can enroll and authenticate on both phones. Both phones use Veridium Authenticator.

If the work phone fails, the user can authenticate with their personal phone. VeridiumID sends push notifications to all enrolled devices. Added devices have these behaviors.

- Any device can be used at any given time, either through PUSH or QR or OFFLINE mode.
- Each device has separate biometric enrollment. Profile modifications occur at a DEVICE level, not a user level. For example, setting FaceID on Device1 does not affect Device2 profile. Each device is treated uniquely for enrollment and profile changes.
- All devices are treated equally for authentication.
- VeridiumID registers a user's multiple devices in the user's account. Administrators manage each device independently. For example you can block or unblock one device and activate or inactivate individual profiles on a device.

Multiple device capability might not be obvious to users so you can notify users of this possibility.

No administrative procedure is needed to use this. Users just download Veridium Authenticator to a second phone and enroll their biometrics.

Providing an Enrollment QR Code

When you have set policies for an integration, users in that integration can enroll and use biometric authentication to access resources.

Users can download the Veridium Authenticator client app to their phones at any time from the Apple Store or Google Play. Users can enroll only after you provide a QR code that you copy from the VeridiumID server integration screen.

Procedure

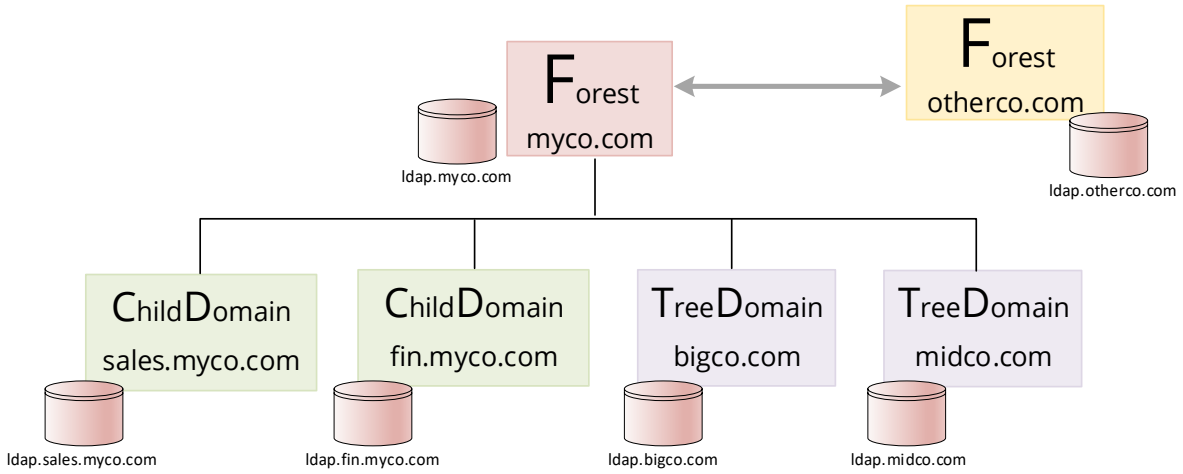
1. Access the VeridiumID Administration Console.
2. Click **Integration**.
3. Find the applicable integration and click **Edit**.
4. Scroll to the QR code for the integration.
5. Copy the QR code image to a file or network location where users can scan it with their Veridium Authenticator app.

Note. Veridium Authenticator downloads policy changes at startup provided it has an internet connection to the server.

LDAP and Active Directory Searches

LDAP hierarchy can be simple. For example, a smaller company might have one LDAP domain. Search a single domain using a call to URL like **LDAP://*LDAP_IPAddress*:389**. The "Configure Active Directory" procedure in *VeridiumID Virtual Appliance Installation* covers this usage.

But commonly over time as companies merge and become more complex, their LDAP hierarchies evolve to reflect the increased complexity. The following diagram shows a complex LDAP hierarchy with two forests.



In the example, the child and tree domains are in the myco.com forest while otherco.com is a separate forest.

VeridiumID uses a single LDAP bind to search for any record in a single forest. The LDAP bind (**ADService.json**, set using the administration console configuration editor) determines the search scope. Here is an example.

```
{
  "LDAPConnections": [
    {
      "URL": "ldap://10.0.12.172:389",
      "baseDN": " DC=sales,DC=myco,DC=com",
      "searchByAttributes": [
        "userPrincipalName",
        "sAMAccountName",
        "objectSid"
      ],
      "attributeNames": {
        "email": "mail",
        "uniqueStaticID": "objectSid",
        "username": "userPrincipalName",
        "mobile": "mobile"
      },
      "credentialsUsername": "bindacct@example.com",
      "credentialsPassword": "MySecretPassword",
      "securityProtocol": "none",
      "securityAuthentication": "simple",
      "sslTrust": "<DEFAULT LDAP TRUST CERTIFICATE>"
```

An LDAP bind has these elements of interest.

baseDN specifies the root distinguished name of a domain to search. This could be a null value if the search scope includes a tree domain.

searchByAttributes are the fields VeridiumID searches in the directory to get the mobile or email contact information.

- These must match what users enter as their username in the VeridiumID client app.
- If more than one attribute is specified, VeridiumID searches all fields on every object.

attributeNames are values needed by VeridiumID operations. For example, to send an OTP (enrollment PIN), VeridiumID needs either mobile or email.

If the mobile number is not in the mobile field (for example, it is in a custom Active Directory attribute named **sms**), you can map the VeridiumID attribute name to the correct Active Directory attribute name. (VeridiumID attribute names on the left map to AD attribute names on the right in this example.)

```
      "attributeNames": {
        "email": "mail",
        "uniqueStaticID": "objectSid",
```

```
        "username": "userPrincipalName",
        "mobile": "sms"
      },
```

mobile and **email** are optional. They are not required for all enrollment policies. Confirm that your enrollment policy does not require OTP to avoid causing enrollment failures. The only required values to map are **uniquestaticID** and **username**.

Searching Multiple Forests

To include another forest in your search perform these tasks:

- Configure trust so the added forest allows searches. See your LDAP documentation for configuring trust between forests.
- Include (contcatenate) an LDAP Bind for the new forest in **ADService.json**.

Here is an example with two LDAP binds. You separate the two binds using a comma. The default **sslTrust** certificate is removed for simplicity.

```
{
  "LDAPConnections": [
    {
      "URL": "ldap://10.40.0.5:3268",
      "baseDN": "",
      "searchByAttributes": [
        "userPrincipalName",
        "sAMAccountName",
        "objectSid",
        "mail"
      ],
      "attributeNames": {
        "email": "mail",
        "uniqueStaticID": "objectSid",
        "username": "userPrincipalName",
        "mobile": "mobile"
      },
      "credentialsUsername": "citrixsa@lab.int",
      "credentialsPassword": "",
      "securityProtocol": "none",
      "securityAuthentication": "simple",
      "sslTrust": "<DEFAULT LDAP TRUST CERTIFICATE>"
    },
    {
      "URL": "ldap://10.40.0.6:3268",
      "baseDN": "",
      "searchByAttributes": [
        "userPrincipalName",
        "sAMAccountName",
        "objectSid",
        "mail"
      ],
      "attributeNames": {
```

```
        "email": "mail",
        "uniqueStaticID": "objectSid",
        "username": "userPrincipalName",
        "mobile": "mobile"
      },
      "credentialsUsername": "admin@vid.local",
      "credentialsPassword": "",
      "securityProtocol": "none",
      "securityAuthentication": "simple",
      "sslTrust": "<DEFAULT LDAP TRUST CERTIFICATE>"
    }
  ],
  "userQueryQuota": 9999999,
  "userQueryQuotaTimeframe": 300,
  "redisURL": "127.0.0.1:6379",
  "redisPrefix": "dev_ldapsrv_",
  "redisPoolsize": 128,
  "serviceAuthenticationCertificates": [
    "8008e86e-9a23-429b-8e77-7b6348811a98",
    "E27B2D93-291C-47F3-BF35-EA6DA590120A",
    "6A79D826-DA26-4B43-8AFF-C3741C32E22C"
  ],
  "jsonSchemaURL": "http://ova.ctxlab.mobi:8080/ADService/rest/configuration/schema"
}
```

Controlling the LDAP Search Scope

Using a single LDAP bind you search either of these scopes:

- Search a single domain by specifying the domain baseDN and port 389 (636 for SSL).
- Search the entire forest (including all child and tree domains) by specifying the LDAP URL for the global catalog level: **LDAP://ldap.myco.com:3268** (3269 for SSL). If the forest includes a tree domain, use a null baseDN (**baseDN:** “”).

Other LDAP binds included in **ADService.json** follow the same rules.

Here is a summary of rules for constructing your **ADService.json** LDAP binds.

LDAP Search Scope	Number of LDAP Binds	Specify Base DN per LDAP bind	LDAP Port to Use
Single domain	1	Yes (Domain root DN)	389/636
Multiple child domains single forest	1	Yes (Forest root DN)	3268/3269
Multiple tree domains single forest	1	No. Use null baseDN: **`baseDN": "",`**	3268/3269

LDAP Search Scope	Number of LDAP Binds	Specify Base DN per LDAP bind	LDAP Port to Use
Multiple domains, child and tree, single forest	1	No. Use null baseDN: `baseDN": "",`	3268/3269
Multiple forests	2 (one for each forest)	Configuration dependent. Use a null baseDN if that forest has a tree domain.	3268/3269

Set LDAP Bind Strings

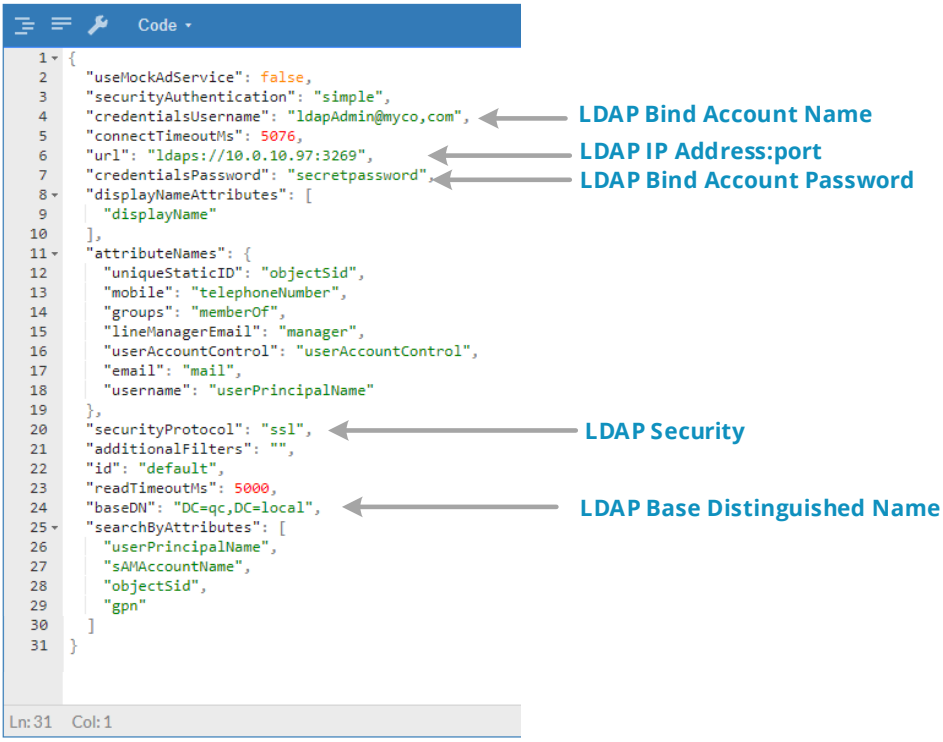
For VeridiumID solutions that use LDAP or Active Directory, you set the LDAP bind when you initially install the VeridiumID server.

You might need to reconfigure LDAP binds when your directory service hierarchy changes as, for example, you add or remove a child or tree domain.

Note. Procedures to set LDAP binds do not apply to VeridiumID solutions that do not use LDAP or Active Directory,

Procedure

1. Access the VeridiumID Administration Console.
2. Click **Configuration** in the Navigation pane.
3. Click **Configuration**.
4. In the Manage Configuration pane, click **Services** > **ldap.json**. The **ldap.json** string opens in the editor.



5. Make any changes, then scroll down and click **Save**.

Handling User Issues

VeridiumID deployments may present conditions where an administrator must block or unblock user accounts or devices or manually remove accounts or devices from the system.

If a user cannot authenticate due to an injury, or mobile device loss or damage, Veridium recommends help desks have procedures to independently verify user identity and grant temporary access by manually authenticating the user.

This table describes some conditions where an administrator can take corrective action to resolve.

Condition	Action
User cannot authenticate due to physical injury or loss of mobile device.	Use authentication override to authenticate the user.
User locked out. Too many failed authentication attempts.	Confirm the user identity using security questions. Then unblock the user account.

Device out of power (needs recharging).	Recharge the phone enough to authenticate. If the need to authenticate is urgent, use authentication override to authenticate the user.
Misplaced mobile device.	Block a misplaced device until the device is found. If the device is found, unblock the device. If a device is confirmed lost, manually remove the user account from the system. Users must register anew when they receive a new device and install the client application. In the meantime, use authentication override to authenticate the user.
Confirmed lost, stolen, or irreparably damaged mobile device.	Manually remove the user account from the system. Users must register anew when they receive a new device and install the client application. Use authentication override to authenticate the user.
Employee leaves the organization.	Manually remove the user account from the system. Remove the user from the identity manager such as Active Directory or Active Directory or LDAP.
Employee receives a new desktop or laptop to which they must authenticate.	Provision the desktop or laptop with the PC login agent.
Employee laptop or desktop is lost or sent for repair.	Block a misplaced device until the device is found. If the device is found, unblock the device. If a device is confirmed lost, manually remove the user account from the system. Users must register anew when they receive a new device and install the client application.

Credentials not accepted when I'm trying to enroll	Confirm credentials are entered correctly with no extra spaces. If other users experience the same issue confirm the server is communicating with the identity server (such as Active Directory. If the Friend certificate has expired, update the friend certificate.
I don't receive my OTP email/SMS	If these channels worked previously, confirm the subscription is still active. Confirm that the enrollment workflow includes an OTP. Check the VeridiumID server configuration for SMTP or SMS (Twilio). Check your log files to determine the server is using these channels without errors. Check Veridium Authenticator log files for incoming SMS or email messages.
OTP not accepted	Make sure you're using the correct OTP with no extra white spaces
Veridium Authenticator app does not launch.	Too many open apps may prevent Veridium Authenticator from launching. Close other open apps. If you haven't enrolled, remove the app following procedures for your phone and reinstall. If you have already enrolled, contact your administrator to remove your account before uninstalling the app and reinstalling.
I don't see an authentication request of my phone when I enter my username on the desktop.	Confirm you have mobile service and that your phone is not in airplane mode.

I get an error when I enter my username on the desktop.	Confirm your desktop is connected to the network. Confirm you have an account on that desktop

Administrator Authentication Override

If a user cannot present their live biometrics due to physical injury, loss of their mobile phone, or a dead phone battery, an administrator can override the authentication mechanism to authenticate the user so he or she can access a resource.

> **Important.** Organizations must have a security policy in place to verify an identity before manually authenticating an affected user. A typical policy consists of security questions that only that individual could answer.

Veridium supports authentication override when an affected user is trying to access their laptop or desktop (the laptop must be on the local network or VPN and can ping the VeridiumID server).

> **Note:** VeridiumID supports authentication override for push notifications where a user enters his or her username. Authentication override is not currently supported for scenarios where users scan QR codes to initiate authentication.

Procedure

1. Independently verify the user's identity according to your security policy.
2. Access the VeridiumID Administration Console.
3. Click **Accounts**.
4. Scroll or search to find the user whose access you want to authenticate.
5. Click the **Active Sessions** tab.
6. Do one of these
   - If the user is trying to access their PC:
     a. Ask the user to click **VeridiumID Authentication** on the PC login screen.
     b. When the new session displays in the Session pane, click 

in the **Actions** column.

- If the user wants to access a resource displaying a Veridium QR code:
  a. Ask the user to scan the displayed QR code.
  b. When the new session displays in the Session pane, click Authenticate in the **Actions** column.

Block, Unblock, or Delete an Account

You can block, unblock, or delete (remove) an account.

Note. Deleting an account removes all biometric templates and other sensitive data belonging to the user that are stored on the server. To remove sensitive data stored on the phone, the user must remove the Veridium Authenticator app from the phone.

Procedure

1. Access the VeridiumID Administration Console.
2. Click **Accounts**.
3. Scroll or search to find the user whose account you want to modify.

3. Click Details on the desired account.
4. Do one of the following:
   a. Block the account.
      - Click  to block the account.
      - Click **Block account** when asked.
   b. Unblock the account.
      - Click  to unblock a blocked account.
      - Click **Unblock account** when asked.
   c. Delete (remove) the account from the system.
      - Click Remove to delete the account.
      - Click **Remove account** when asked.

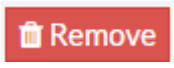
Block, Unblock, or Delete a Device

You can block and unblock a device.

Procedure

1. Access the VeridiumID Administration Console.
2. Click **Accounts**.
3. Scroll or search to find the user whose device you want to modify.
4. Click 

on the desired account.
5. Click **Devices** in the Account Details pane.
6. Scroll down to the desired device and click 

.
7. Do one of the following:
   a. Block the device.
      - Click  to block the device.
      - Click **Block device** when asked.
   b. Unblock the device.
      - Click  to unblock a blocked device.
      - Click **Unblock device** when asked.
   c. Delete the device from the system.
      - Click 

to delete the device.
      - Click **Remove device** when asked.

Creating a New Device

The only situation where an administrator manually creates a device is when adding administrator privileges to an enrolled user's account. See "Add Administrative Privileges to Enrolled User Accounts'.

VeridiumID automatically creates proper devices when users enroll their Veridium client applications or Windows Veridium desktop software.

Veridium Authenticator Logs

In case Veridium Authenticator issues are difficult to diagnose, users can enable and export Veridium Authenticator logs using basic, verbose, and debug logging levels. See *Veridium Maintenance* for details.

Monitor System Activity

System Health Monitoring

Administrators can monitor system health by tracking the performance metrics of VeridiumID platform subsystems.

Health metrics include:

cassandra.response.time	The current cassandra response time in milliseconds. Default threshold is 50 milliseconds.[1]
kafka.response.time	The current kafka response time in milliseconds. Default threshold is 50 milliseconds.[1]
system.jvm.memory.free	System jvm free memory in mb
system.jvm.memory.max	System jvm maximum memory in mb
system.jvm.memory.total	System jvm total memory available in mb
system.jvm.memory.used	System jvm used memory in mb
zookeeper.response.time	The current zookeeper response time in milliseconds. Default threshold is 50 milliseconds.[1]

[1] You can change the default setting in **metrics.json** in the dashboard Configuration page.

Logs and Reports

VeridiumID provides these logs for monitoring system activity. Use these logs for forensic analysis of system use.

- **History** – Logs creation, modification, and deletion of accounts and devices as well as every authentication using these event flags.
    - ACCOUNT_REGISTER
    - ACCOUNT_REMOVE
    - ACCOUNT_UPDATE_STATUS
    - ACCOUNT_UPDATE_GROUPS
    - SESSION_FINISH
    - SESSION_REGISTER
    - DEVICE_UPDATE
    - DEVICE_REPLACE
    - DEVICE_REMOVE

- DEVICE_INSERT

- **Sessions** – Records authentication sessions and status (completed, failed, or cancelled). Every authentication, whether for system access or signing a banking transaction, is signed and recorded in the Sessions log.

- **Action Logs** – Logs each call to the web services API from external client applications. All parameters are captured along with the call.

  For a complete listing of web service APIs that may be called, see the following VeridiumID server url:

- **Reports** – Returns statistics for time frames you specify for these facilities:

Authentication Sessions Report	For each application, returns the number of authentications attempted and the number that failed. or succeeded, or were cancelled for each day included in the range of dates.
Groups Definition Report	Returns the current configuration of groups, roles and permissions.
Sessions Times Report	For each session established with the VeridiumID server in a specified date range, lists the type of session parameters identifying the user and device involved in the session. • **QR** an authentication session started by scanning a QR code • **PUSH** an authentication session started using a push notification. • **MOBIL** an authentication session to authorize a change to the mobile application such as disabling a biometric. • **PROXY** an authentication session generated from a proxy server such as a local phone app. • **EXTERNAL_TOKEN** any authentication involving an external token like an OTP or PIN. **Procedure** 1. Choose a fixed or custom time period. Then click **Apply**. 2. Click **Export CSV** to generate and download the report for viewing.

License Compliance Monitoring

You can monitor your license usage by running monthly usage reports. Veridium recommends running these reports on the 1st day of a month and choosing a range of all days in the previous month.

Procedure

1. Access the VeridiumID Administration Console.
2. Click **Reports** so reports are visible in the navigation pane.
3. Click **Authentication Sessions Report**.
4. Click the Period time bar 

.
5. Choose or enter the appropriate range.
6. Click 

. After a minute, the report displays.
7. Click 

. The PDF report downloads to your desktop.
8. Click **Reports** again so reports are visible in the navigation pane.
9. Click **Accounts Allocation Report**.
10. Click 

. After a minute, the report displays.
11. Click 

. The PDF report downloads to your desktop.
12. Save the PDF reports for reference.

Back Up and Restore the VeridiumID Database

You protect biometric data and account data by backing up the VeridiumID database. VeridiumID uses an Apache Cassandra database.

Cassandra nodes are deployed as a cluster in your data center. Each node in the cluster is a peer of the other nodes containing all the data for your VeridiumID deployment. Data is maintained in one or more keyspaces (namespaces) as appropriate for your Veridium environment.) Cassandra protocols handle synchronization across all nodes in the cluster.

> **Note.** To deploy VeridiumID globally you add nodes to your cluster in other data centers with VeridiumID servers. Cassandra protocols work across separate data centers to keep data synchronized in a global cluster.

Veridium support personnel configure automatic backup of each Cassandra database node when the system is installed. A cron job triggers regular snapshots (typically one per day) of your database while the system is online, saving them to a storage location that you designate. The snapshot first flushes all in-memory writes to disk, then makes a hard link of the SSTable files for each keyspace, ensuring a full restore of all backed up data.

If a Cassandra node goes offline for less than three hours, it automatically resynchronizes with the other nodes when it returns online. After three hours offline, an administrator must perform a **nodetool repair** recovery operation.

The **nodetool** program is a standard Cassandra utility that supports these operations:

- Take snapshots.
- Delete snapshots.
- Manually resynchronize a node.
- Restore a node from a snapshot.

Take Snapshots

Cassandra nodes are configured for automatic snapshots, so you do not normally need to take manual snapshots.

If you need to take a snapshot for some reason such as anticipating an offline condition, you can take one individually on a node using the **nodetool snapshot** command.

Snapshots are saved to ***data_directory_location/keyspace_name/table_name-UUID*/snapshots/*snapshot_namedirectory***. Each snapshot directory contains numerous ***.db** files that contain the data at the time of the snapshot.

Procedure

- Issue this nodetool command:

  ```
  nodetool -h localhost -p 7199 snapshot mykeyspace
  ```

Delete Snapshots

Individual snapshots do not occupy a large amount of space but can accumulate over time. You manually delete unneeded snapshots to avoid exhausting disk space.

Procedure

- Clear an individual snapshot using this **nodetool** command:

  ```
  nodetool -h localhost -p 7199 clearsnapshot -i snapshot_name
  ```

  Clear all snapshots using this **nodetool** command:

  ```
  nodetool -h localhost -p 7199 clearsnapshot
  ```

Manually Resynchronize a Node

If a node has been offline for more than three hours due to a network outage or was shut down but has not been corrupted or lost data, use this procedure to manually resynchronize the data with other servers in the cluster.

Procedure

- Manually resynchronize a node using this command on the node when it is back online:

  ```
  nodetool repair
  ```

Restore a Node from a Snapshot

If a node has lost data or data has been corrupted, restore it to the state of a specific snapshot using this procedure.

Procedure

1. Run **nodetool drain**.
2. Shut down the Cassandra service using this command:

   ```
   service ver_cassandra stop
   ```

3. Clear all files in the **/vid-app/1.0.0/cassandra/commitlog** directory.
4. Delete all ***.db** files in the following directory:
   data_directory_location/keyspace_name/keyspace_name-table_name.

   Important: Do not delete the **/snapshots** and **/backups** subdirectories.

5. Locate the most recent snapshot folder in this directory:

   data_directory_location/keyspace_name/table_name-UUID/snapshots/snapshot_name

6. Copy the folder contents into this directory:

   data_directory_location/keyspace_name/table_name-UUID

7. Restart the node.

   ```
   service ver_cassandra start
   ```

8. Run **nodetool repair**.

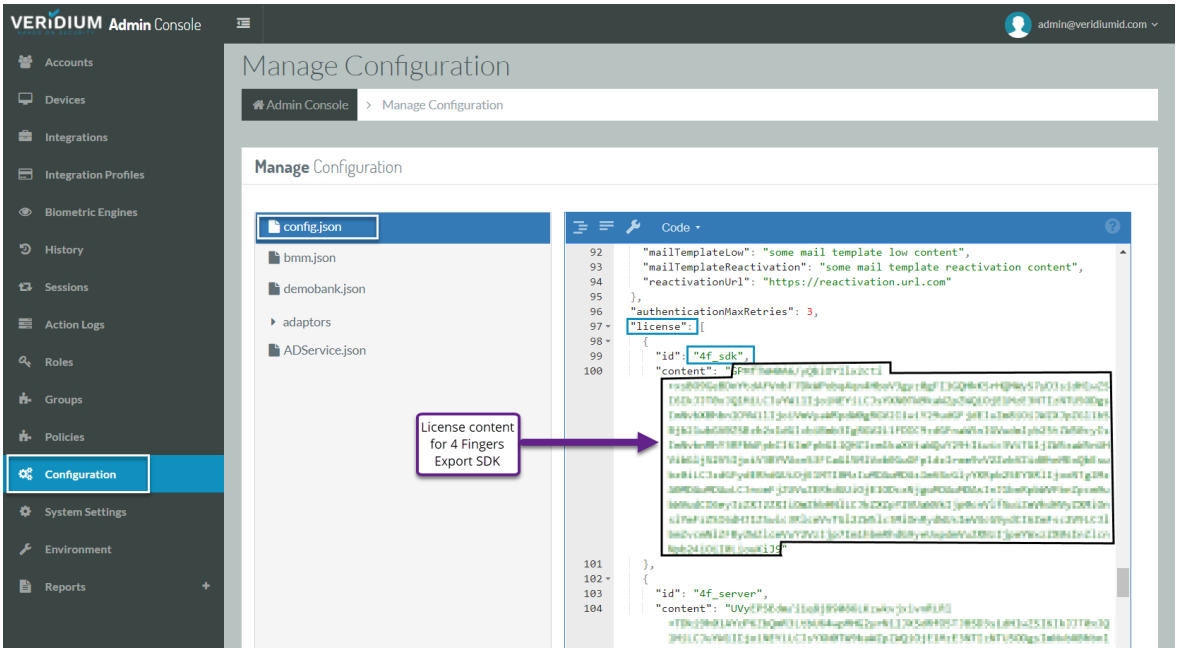
Update VeridiumID Licenses

You must update the VeridiumID server licenses under the following conditions.

- when licenses expire
- when licenses become invalid such as when you increase capacity beyond current licensed limits
- when the server SSL certificate expires or is replaced as the licenses are bound to the certificate

Procedure

1. Access the VeridiumID Administration Console.
2. Log in to the VeridiumID Server GUI using your browser.
3. Click **Configuration** in the Navigation pane.
4. Click **config.json**.
5. In the **Code** pane, scroll down to the license section.
6. Paste the content of each license into the corresponding license property.



7. After copying all licenses, scroll down and click **Save**.

Generate the Super Administrator Certificate

If you lose access to the Administration dashboard due to an expired or lost administrator certificate, you can issue commands to generate a replacement super admin certificate.

Procedure

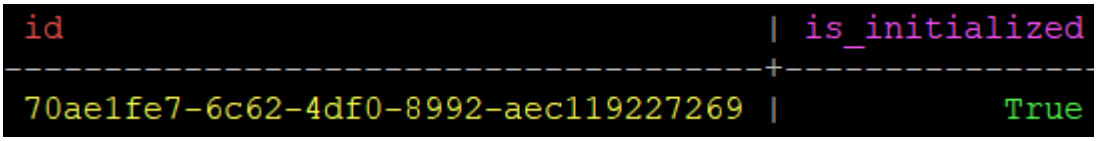
1. Log in to the VeridiumID command line using PuTTy.
2. Access Cassandra using CQLSH:

   `/opt/veridiumid/cassandra/bin/cqlsh *host*

   Where: *host* is the IP address or the FQDN on which Cassandra listens. For example, in a POC deployment the host IP might be 127.0.0.1.
3. Issue the following CQL query:

   `select * from veridium.system_status;`

   The database returns a response like the following:

4. Issue this query to set the is_initialized field to False:

   `update veridium.system_status set is_initialized=False where id='*ID*';`

   Where: *ID* is the id value returned from the database.
5. Generate the new certificate. In the terminal use the following command:

   `curl http://host:8088/websec/rest/setup/setupInitialData?exportpath=/tmp`

   Where *host* is the IP address or FQDN on which Tomcat listens. For example, in a POC deployment the host IP might be 127.0.0.1.

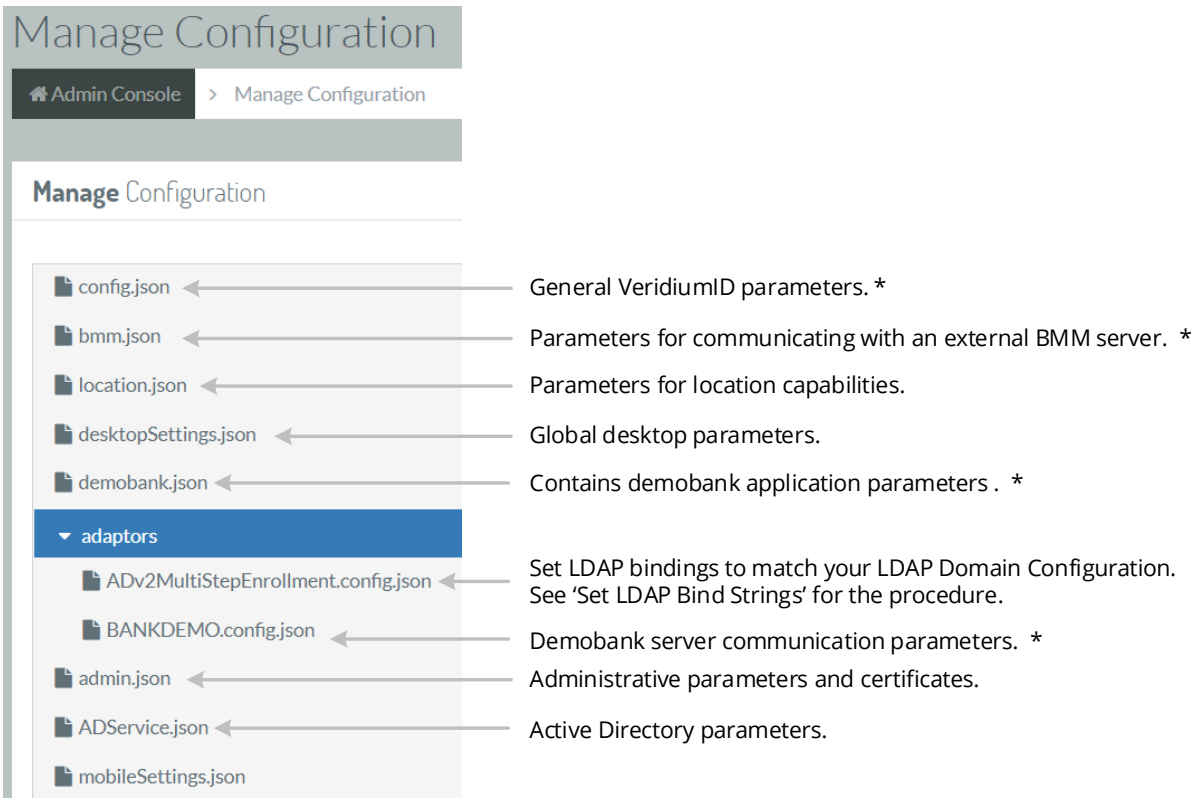
   VeridiumID generates the new certificate in **/tmp**.
6. Use the default certificate password to import the certificate into your browser:

   FAEE2045-2B0C-4725-9233-E46F3FF307C0.

View and Manage Configuration Settings

VeridiumID server operations rely on sets of configuration values like pathnames, values, certificates, and other parameters needed for server operations.

Parameters are expressed as JSON strings associated with specific server apps and services. VeridiumID includes an editor to view or modify json strings.



* Your Veridium sales and support team sets these parameters as needed.

Procedure

1. Access the VeridiumID Administration dashboard.
2. Click **Configuration**.
3. Use the page navigation to find the configuration setting you want to view or change.
4. Click a string name to view or edit the string in the **Code** pane.
5. Click **Save** to save your changes.
6. Take appropriate action to confirm your server is using the new values.

Appendix A Console Interfaces

Administrators use the console to view statics, history events, and reports, and to manage users, devices, and server settings.

This table summarizes the use of each console interface. For usage details see the task procedures in this guide.

Console Operation	Description
Dashboard	The dashboard displays: • The number of successful and failed authentications for the last 31 days. You can modify the date range • For each biometric engine, the number of successful and failed authentications.
Accounts	An account defines a single user with these parameters: **Account name** that identifies a user. This may be an email address. The system creates an account ID that uniquely identifies that user account as a user may have multiple accounts. For example, a user may be a default user who has enrolled with their smart phone to use VeridiumID authentication. That same user may be a VeridiumID server administrator and has a separate account for that purpose. **External ID** that maps to a user's name as contained in a third-party application like Active Directory. **Account type** that maps to an Integration name. By default, users are members of the Guest group that allows enrollment and authentication. You edit a user account to add or remove Groups from the account.
Devices	Displays devices enrolled in the system. A device is any component that interacts with the VeridiumID server such as a mobile device, a laptop, a tablet, or browser.
Integrations	An **integration** shows parameters integrating client devices with your organization's authentication workflows. An organization can have multiple integrations, one for each system you are integrating with.

Console Operation	Description
	Integrations define risk conditions at authentication time based on user actions or behavior. The server calculates these at sign-on to apply appropriate authentication requirements such as pin only, or facial, or use of multiple biometric methods. For example, the server may detect sign-on from an unusual user IP address and require additional biometric authentication methods. You can also define enrollment parameters that integrate with your business workflows. For example, you can specify use of SMS or email verifications during enrollment to confirm a user is in possession of a specific device or email account. **Note:** The **Integrations** screen is a read-only interface. The values in this screen are taken from configuration parameters in the business adaptor.
Integration Profiles	An integration profile associates a user with a specific integration and the adapter that interacts with the third-party server or system such as Active Directory.
Biometric Engines	This read-only interface shows the biometric engines configured for your VeridiumID server.
History	View all events that occurred in the system including enrollments, authentications, session initiations and terminations.
Sessions	Provides records of all authentication operations in the system.
Action Logs	Logs each use of the VeridiumID Server API which handles enrollment, authentication and other requests from mobile clients and other external devices.
Roles	Use these operations to view system roles and to create, edit, and remove custom roles.
Groups	Use these operations to view system groups and to create, edit, and remove custom groups.

Console Operation	Description
Policies	Set a group policy to specify a subset of Available Biometric Methods and override what is set in Required Biometric Methods.
Configuration	View and change parameters for services like adaptor bindings and SAML. Set LDAP bindings to match your domain configuration. See *VeridiumID Administration* and various configuration guides for specific procedures. View and Edit Email, LDAP and SMS services. View and renew truststore certificates. Validate friend certificates for VeridiumID components . When certificates expire you can regenerate (and install) new friend certificates.
Environment	Shows the VeridiumID server version, the software build version, and current memory and disk usage.
Reports	Generate, preview, and export as PDF, reports about applications, authentications sessions, daily activations, group definition, and account allocation and usage.

Appendix B. Understanding Accounts and Integrations

Accounts and integrations in the Administration UI rely on several identifiers to maintain proper associations among accounts, user devices, and integrations. The identifiers appear in administration interfaces. Normally you do not need to use these identifiers as they are used more for problem resolution by Veridium product support personnel.

An account is a collection of devices, privileges, integrations, and activity records associated with a Veridium client app providing a comprehensive view of relevant objects and activity. An account has an **accountID** that uniquely identifies it in the system.

After enrolling, an end user's phone has a **deviceID** and a certificate bound to the account so that VeridiumID can find and communicate with the phone.

An account can have multiple devices such as a mobile phone hosting a client app and a desktop PC where a user tries to sign on by entering a password. Each device has its own unique **deviceID**.

An account binds to an integration using an **integration profile** which contains the **accountID** and a **memberID**. The **memberID** is the **ID** of the integration as represented in the profile, meaning the account is a member of the integration. An account can belong to multiple integrations with different integration profiles for each membership.

Finally, identifiers have a human-readable name displayed nearby to help you find data items in the interface.

Appendix C. Groups, Roles and Permissions

This appendix describes system groups, system roles, and permissions.

Groups

Groups provide a scalable way to add sets of permissions to different administrative users.

System groups are fixed and cannot be edited or deleted. You can create custom groups with custom roles if you need specialized sets of permissions. In most cases, the default groups are adequate for access control needs.

This table describes the system groups.

System Group	Roles	Permissions
Administrators The group of admin users	admin, active	Default client access, Default user, Cross Application Administrators
Alerts The group of alerts users	alerts, admin	Cross Application Administrators, Alerts administrators
AppAdmins The group of application administrators	appadmin	Application Administrators
AutoTest The group of automatic test machines	admin, autotest	Cross Application Administrators
Finance The group of finance users	reports	User can load reports
Guests The group of guests users	default	Default user
Sales The group of sales users	reports	User can load reports

System Group	Roles	Permissions
SystemAdmins The group of administrators who can change system settings and configuration	sysconfig	System Administrators
TechSupport The group of technical support users	techsupport	Technical Support
Tester The group of testers	tester,admin	Run test, Cross Application Administrators
Users The group of enrolled client application users	active	Default client access, Default user

Roles

Roles are collections for one or more permissions. System roles are fixed and cannot be edited or deleted. You can create customized roles if you need specialized sets of permissions.

This table describes the system roles.

System Role	Permissions
active	Default client access, Default user
admin	Cross Application Administrator
alerts	Alerts Administrators
analyst	Analyze Data
appadmin	Application Administrator
autotest	The user role used by automatic tests
default	Default user
reports	User can load reports

System Role	Permissions
sysconfig	System administrators (The User Role that can change system settings and configuration)
techsupport	Technical Support (The User Role that can delete accounts and devices)
tester	Run tests

Permissions

Permissions are the basic unit of access, controlling access to specific object types in a VeridiumID server.

The following table describes available default permissions.

Permission Name	Objects Protected
Default client access	access account create and account remove operations.
Default user	access operations that compare biometric templates against CBVs (current biometric vectors).
Cross Application Administrator	access objects related to all client applications in the system (accounts, devices, integrations, profiles, history) .
Alerts Administrators	View alerts.
Analyze Data	View data objects.
appadmin	Access all objects for a single application.
autotest	Run automatic tests.
User can load reports	Run and view reports.
System administrators	Access system settings and configuration (functions in the navigation panel)
Technical Support	Manage accounts and devices.
Run tests	Run tests using controls in the navigation panel.

Other Permissions

- License Administrators
- Certificate Administrators
- Configuration settings administrators
- ACL Administrator
- Run configuration settings
- Write license agreement
- Write configuration settings
- Read configuration settings
- Read license agreement
- Accounts administrators
- Permissions administrators
- Members profiles administrators
- Devices administrators
- Mobile settings administrators
- Accounts sites administrators
- Early warning system (EWS) administrators
- Run mobile settings
- Execute EWS job
- Execute accounts jobs
- Write mobile settings
- Write Accounts
- Write members profiles
- Write devices
- Write permissions
- Write alerts
- Write EWS data
- Write accounts sites
- Read mobile settings
- Read accounts sites
- Read devices
- Read EWS data
- Read alerts
- Read permissions
- Read members profiles
- Read accounts
- System logging
- System history
- Users sessions

Appendix D. Configure Email Notifications

You can configure email notifications to users and line managers for enrollment and authentication events. You can use placeholders (variables) to include event details that are relevant to the user.

Configure email notifications for these events:

REGISTRATION_USER_NOTIFICATION: Notice to users and other desired recipients.

REGISTRATION_LINE_MANAGER_NOTIFICATION: Notice to managers and other desired recipients.

SESSION_SUCCESS: Notice to users and other desired recipients.

Prerequisite

Email notifications require access to a user's email address. For Directory Service integrations, VeridiumID searches LDAP for the user's email attribute. For other integrations, users must enroll using their email address to register their account.

Procedure

1. In the administration dashboard, click **Notifications**.
2. Click **Edit** for the notification you want to edit.
3. Slide **Enabled** switch to **On**, if not enabled already.
4. Enter Recipients if different from the default.
5. Enter the **Subject** and **Body** content.

   The Subject and Body fields support placeholders (variables) which insert text strings associated with the user's registered account, integration profile or device.

   The Body field supports html formatting using the Apache FreeMarker™ template engine. See the Apache FreeMarker™ manual for more details.
6. Click **Validate body** to confirm proper html coding.
7. Click **Save** to exit the editor and save your changes.

Appendix E. Configure Push Notification and SMS Services

Push notifications handle authentication scenarios where users enter their usernames in an application to start biometric authentication workflows.

Some enrollment scenarios send SMS PIN codes to user phones to verify a user has the expected phone. Users enter received PIN codes into Veridium Authenticator during enrollment.

Use these procedures to configure these services.

Configure APNS and FCM Notification Services

If your workflows involve push notifications, you must set up Apple Push Notification service (APNs) and Google Cloud Messaging (GCM), now called Firebase Cloud Messaging (FCM) for Android phones.

Contact these services directly to subscribe and receive subscription parameters:

- APNS certificate
- APNS certificate password
- FCM Key (Android)

Procedure

1. In the administration dashboard, click **Configuration>Configuration**>.
2. Click **config.json**.
3. Enter these bolded values into the parameters for apns and gcm:

```
"apns": [
  {
    "cert": "<APNS certificate> ",
    "id": "com.veridiumid.authenticator",
    "production": true,
    "pwd": "APNS certificate password"
  }
],

"gcm": [
  {
    "id": "com.veridiumid.authenticator",
    "key": "FCM key"
```

```
        }
    ],
```

4. Scroll down and click **Save** to continue.

Configure SMS for Sending PIN Codes

If your enrollment protocol sends SMS pin codes for device verification to complete enrollment, configure SMS which uses the Twilio SMS service.

Contact Twilio directly to subscribe and receive these SMS parameters:

- SMS service account ID
- SMS service password
- SMS service phone number(s)

Procedure

1. In the administration dashboard, click **Configuration>Configuration**>.
2. Click **adaptors** > **ADv2MultiStepEnrollment.config.json**.
3. Enter these bolded values into the parameters for apns and gcm:

   ```
   "smsServicePassword": "SMS service password",
   "smsServiceAccountID": "SMS service account ID",
   "smsServicePhonenumbers": "+SMS service phone number",
   ```

   If you have multiple Twilio-issued phone numbers, separate them with a comma as shown here:

   ```
   "smsServicePhonenumbers": "+12345678901,+134567989012",
   ```

4. Scroll down and click **Save** to continue.